

# Distributed bounded-error parameter and state estimation in networks of sensors

Michel Kieffer

► **To cite this version:**

Michel Kieffer. Distributed bounded-error parameter and state estimation in networks of sensors. Cuyt, A. and Krämer, W. and Luther, W. and Markstein, P. Numerical validation in current hardware architectures – From embedded system to high-end computational grids, 0, Springer-Verlag, 2009. <inria-00420955>

**HAL Id: inria-00420955**

**<https://hal.inria.fr/inria-00420955>**

Submitted on 30 Sep 2009

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Distributed bounded-error parameter and state estimation in networks of sensors

Michel Kieffer<sup>\*1</sup>

LSS - CNRS - SUPELEC - Univ Paris-Sud, Plateau de Moulon, 91192  
Gif-sur-Yvette, France,  
kieffer@lss.supelec.fr,  
WWW home page: <http://michel.kieffer.lss.supelec.fr>

**Abstract.** This paper presents distributed bounded-error parameter and state estimation algorithms suited to measurement processing by a network of sensors. Contrary to centralized estimation, where all data are collected to a central processing unit, here, each data is processed locally by the sensor, the results are broadcasted to the network and taken into account by the other sensors. A first analysis of the conditions under which distributed and centralized estimation provide the same results has been presented. An application to the tracking of a moving source using a network of sensors measuring the strength of the signal emitted by the source is considered.

## 1 Introduction

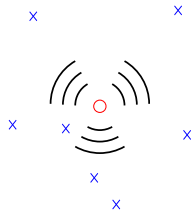
A wireless sensor network (WSN) consists of spatially distributed autonomous devices equipped with sensors and interconnected via wireless links. Sensors may be designed for measuring pressure, temperature, sound, vibration, motion... Initially WSN were developed for military applications (battlefield surveillance). Now, many civilian applications (environment monitoring, home automation, traffic control) may take advantage of WSN, see, *e.g.*, [1, 2].

Applications suggest many research topics, such as the design of protocols for communication between sensors, localization problems, data compression and aggregation, security issues... All these problems are made more complicated by the constraints imposed on each node of the WSN, which usually has limited computing capabilities, communication capacity, and, to increase its autonomy, has strong power consumption constraints.

The application considered here is WSN for source tracking, which may be important when considering mobile phone localization and tracking, computer localization in an ad-hoc network, co-localisation in a team of robots, speaker localization... Figure 1 illustrates a typical localization problem: a source represented by a circle moves in a field of sensors, each of which is represented by a cross.

---

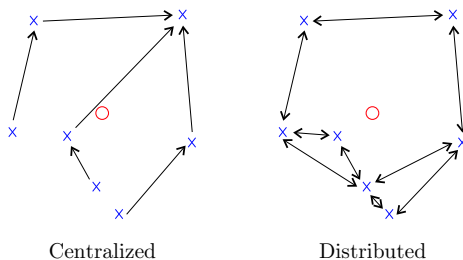
\* This work has been partly supported by the NoE NEWCOM++



**Fig. 1.** Source (o) and sensors (x)

The localization technique used depends on the type of information available to the sensor nodes. Time of arrival (TOA), time difference of arrival (TDOA) and angle of arrival (AOA) usually provide the best results [3], however, these quantities are difficult to obtain, as they require a good synchronization between timers (for TOA), exchanges between sensors (for TDOA) or multiple antennas (for AOA). Contrary to TOA, TDOA or AOA data, readings of signal strength (RSS) at a given sensor are easily obtained, as they only require low-cost sensors or are already available, as in IEEE 802.11 wireless networks, where these data are provided by the MAC layer [4].

This paper focuses on source localization and tracking from RSS data. Centralized approaches (see Figure 2, left) have been proposed to solve this problem for acoustic sources [5] and for sources emitting electromagnetic waves, see, *e.g.*, [6–8]. In the first case, some knowledge of the decay rate of the RSS (*path loss exponent*) is needed for efficient nonlinear least squares estimation. In the second case, an off-line training phase is required to allow maximum *a posteriori* localization. In both cases, a good initial guess of the location of the source facilitates convergence to the global minimum of the cost function. Distributed approaches (see Figure 2, right) have also been employed, *e.g.*, in [9], where a distributed version of a nonlinear least squares solver has been presented. When badly initialized, it suffers from the same convergence problems as the centralized approach, as illustrated in [10], which advocates projection on convex sets. Nevertheless, the latter requires an accurate knowledge of the source signal strength and of the path loss exponent.



**Fig. 2.** Centralized (left) and distributed (right) processing of measurements

The localization and tracking problems are considered as distributed discrete-time state estimation problems involving bounded state perturbations and measurement errors. This problem is addressed with the help of interval analysis [11, 12], which will provide at each node of the network and at each time instant a set estimate guaranteed to contain the true location of a moving source, provided that the hypotheses on the model and measurement noise are satisfied. Section 2 describes an idealized and a practical distributed state estimation algorithm able to deal with bounded-error measurements. Section 3 presents the application of the preceding algorithm to source localization and tracking.

## 2 Distributed state estimation

Consider a system described by a discrete-time state equation

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{w}_k, \mathbf{u}_k), \quad (1)$$

where  $\mathbf{x}_k$  is the state vector of the model at time instant  $k$  (the sampling period is  $T$ ). The state perturbation vector  $\mathbf{w}_k$  accounts for unmodelled parts of the system and is assumed to remain in a known box  $[\mathbf{w}]$ . The input vector  $\mathbf{u}_k$  is also assumed known. At  $k = 0$ ,  $\mathbf{x}_0$  is only assumed to belong to some (possibly large) known set  $\mathbb{X}_0$ .

Assume that at time  $k$ , each sensor  $\ell = 1 \dots L$  of a WSN has access to a noisy measurement vector  $\mathbf{y}_k^\ell$ . The measurement process is described by the observation equations

$$\mathbf{y}_k^\ell = \mathbf{g}_k^\ell(\mathbf{x}_k, \mathbf{v}_k^\ell), \quad (2)$$

where  $\mathbf{v}_k^\ell$  is the measurement noise, assumed bounded in some known box  $[\mathbf{v}]$ . Usual observation equations are

$$\mathbf{g}_k^\ell(\mathbf{x}_k, \mathbf{v}_k^\ell) = \mathbf{h}_k^\ell(\mathbf{x}_k) + \mathbf{v}_k^\ell \quad (3)$$

or

$$\mathbf{g}_k^\ell(\mathbf{x}_k, v_k^\ell) = \mathbf{h}_k^\ell(\mathbf{x}_k) \cdot v_k^\ell, \quad (4)$$

depending on whether the measurement noise is additive or multiplicative.

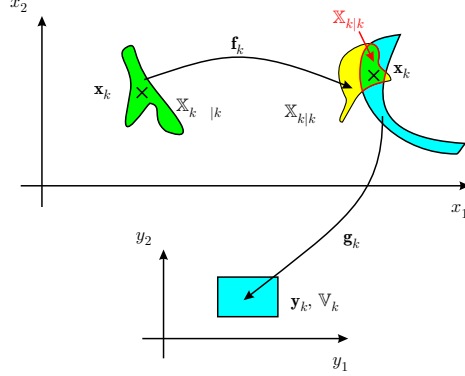
### 2.1 Back to centralized discrete-time state estimation

Centralized state estimation is briefly summarized, since it constitutes the reference which distributed algorithms should reach.

When all measurements at time  $k$  are available at a central processing unit, one gets

$$\begin{cases} \mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{w}_k, \mathbf{u}_k), \\ \mathbf{y}_k = \mathbf{g}_k(\mathbf{x}_k, \mathbf{v}_k), \end{cases} \quad (5)$$

with  $\mathbf{y}_k^\top = \left( (\mathbf{y}_k^1)^\top, \dots, (\mathbf{y}_k^L)^\top \right)$  and  $\mathbf{v}_k^\top = \left( (\mathbf{v}_k^1)^\top, \dots, (\mathbf{v}_k^L)^\top \right)$ . Determining an estimate for  $\mathbf{x}_k$  from the measurement  $\mathbf{y}_\ell$ ,  $\ell = 0 \dots k$  is a classical state



**Fig. 3.** Idealized recursive bounded-error state estimator

estimation problem, the solution of which depends on the linearity of (1) and (2) and on the noise model. For a gaussian noise, with linear state and observation equations, the Kalman filter [13] is the natural solution. When the model is non-linear, one may use an extended Kalman filter [14], gridding techniques [15], or particle filters [16]. In a bounded-error context, with a linear model, the set of state vectors consistent with the model and noise on the measurements may be evaluated exactly using polytopes [17], or outer-approximated using ellipsoids [18]. With a nonlinear model, again, an outer-approximation of the state is possible using subpavings, *i.e.*, unions of non-overlapping boxes [19].

Summarizing the information available at time  $k$ , one gets

$$\mathcal{I}_k = \left\{ \mathbb{X}_0, \{[\mathbf{w}_j]\}_{j=1}^k, \{[\mathbf{v}_j]\}_{j=1}^k, \{[\mathbf{y}_j]\}_{j=1}^k \right\}. \quad (6)$$

Centralized bounded-error state estimation at time  $k$  aims at characterizing the set  $\mathbb{X}_{k|k}$  of all values of  $\mathbf{x}_k$  that are consistent with (1), (2), and  $\mathcal{I}_k$ . One may propose an idealized algorithm [19], alternating, as the Kalman filter a prediction step involving (1)

$$\mathbb{X}_{k|k-1} = \left\{ \mathbf{f}_k(\mathbf{x}, \mathbf{w}, \mathbf{u}_k) \mid \mathbf{x} \in \mathbb{X}_{k-1|k-1}, \mathbf{w} \in [\mathbf{w}] \right\} \quad (7)$$

and a correction step accounting for the new measurement using (2)

$$\mathbb{X}_{k|k} = \left\{ \mathbf{x} \in \mathbb{X}_{k|k-1} \mid \mathbf{y}_k = \mathbf{g}_k(\mathbf{x}, \mathbf{v}), \mathbf{v} \in [\mathbf{v}]^L \right\}. \quad (8)$$

The two steps of the idealized algorithm are depicted in Figure 3.

This idealized algorithm requires the evaluation of the direct image of a set by a function in the prediction step (7) and the evaluation of the inverse image in the correction step (8).

Usually, state estimation starts with an observability study to determine whether there is a chance to get an satisfying state estimate [20, 21]. With set estimators, this study is not required *a priori*. A lack of observability typically

results in the increase of the size of the components of  $\mathbb{X}_{k|k}$  which are not observable. Alternatively,  $\mathbb{X}_{k|k}$  may also consist of several disconnected subsets. Lack of observability may thus be detected during the estimation process.

## 2.2 Distributed state estimation

Distributed versions of the Kalman filter have been proposed in [22], assuming linear models, gaussian noise, and instantaneous communications. Application to distributed estimation in power systems have been addressed in [23] and to distributed estimation in WSN are considered in [24]. Nevertheless, to the best of our knowledge, no similar tools have been proposed in a bounded-error context.

Consider a network of  $L$  sensors. Ideally, any sensor  $\ell$ ,  $\ell = 1 \dots L$  of the WSN should provide

$$\mathbb{X}_{k|k}^\ell = \mathbb{X}_{k|k}. \quad (9)$$

To establish conditions under which (9) is satisfied, some notions of graph theory have to be recalled. For more details, the reader is referred to [25, 26].

The network of  $L$  sensors is represented by a *graph*  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ .  $\mathcal{V}$  is the set of  $L$  *vertices* of the graph, each vertex representing a sensor of the network and  $\mathcal{E}$  is the set of *edges* of the graph. An edge  $\{k, \ell\} \in \mathcal{E}$  connecting two vertices  $k \in \mathcal{V}$  and  $\ell \in \mathcal{V}$  indicates that the two corresponding sensors are able to directly exchange information; the graph is thus *undirected*. In what follows, it is assumed that  $\mathcal{G}$  is *entirely connected*, *i.e.*, that there is always a path from any vertex to any other vertex in  $\mathcal{G}$  and that each vertex is connected to itself.

The *distance* between two vertices in  $\mathcal{G}$  is the number of edges in a shortest path connecting them. Consider a vertex  $\ell \in \mathcal{V}$ , then

$$\mathcal{C}(\{\ell\}) = \{k \in \mathcal{V} \mid (k, \ell) \in \mathcal{E}\} \quad (10)$$

denotes the set of all vertices that are directly connected to  $\ell$ , *i.e.*, that are at a distance not larger than one of  $\ell$ . More generally, for any  $\mathcal{W} \subset \mathcal{V}$ ,  $\mathcal{C}(\mathcal{W}) \subset \mathcal{V}$  is the set of all vertices which are at a distance not larger than one from a given vertex of  $\mathcal{W}$ . The set

$$\mathcal{C}(\mathcal{C}(\{\ell\})) = \mathcal{C}^2(\{\ell\}) \quad (11)$$

contains thus all vertices that are at a distance not larger than two of  $\ell$ . More generally,  $\mathcal{C}^n(\{\ell\})$  contains all vertices that are at a distance not larger than  $n$  of  $\ell$ . The *eccentricity*  $\varepsilon$  of a vertex  $\ell \in \mathcal{V}$  is the largest distance between  $\ell$  and any other vertex in  $\mathcal{G}$ . Finally, the *diameter*  $d$  of  $\mathcal{G}$  is the maximum eccentricity of any vertex in  $\mathcal{G}$ .

**Hypotheses and idealized algorithm.** The following measurement processing and communication will be considered. At time  $k$ , each sensor processes its own measurement  $\mathbf{y}_k^\ell$ . Between time  $k$  and  $k + 1$ , a first round trip is considered ( $r = 1$ ) in which each sensor  $\ell$  broadcasts its own estimate  $\mathbb{X}_{k|k}^{\ell,r}$  to all the sensors of the network (only those which are directly connected to  $\ell$  receive the information). Then each sensor  $\ell$  receives and processes  $\mathbb{X}_{k|k}^{s,1}$ ,  $s \in \mathcal{C}(\{\ell\})$ . Depending on

the sampling time  $T$ , more round trips ( $r > 1$ ) may be considered. Just before time  $k + 1$ , each sensor  $\ell$  builds a final estimate  $\mathbb{X}_{k|k}^\ell$ .

This way of processing and transmitting information leads to the following idealized distributed algorithm.

For each sensor  $\ell = 1 \dots L$ ,

1. At time  $k$ :

$$\mathbb{X}_{k|k-1}^\ell = \left\{ \mathbf{f}_k(\mathbf{x}, \mathbf{w}, \mathbf{u}_k) \mid \mathbf{x} \in \mathbb{X}_{k-1|k-1}^\ell, \mathbf{w} \in [\mathbf{w}] \right\}. \quad (12)$$

$$\mathbb{X}_{k|k}^{\ell,0} = \left\{ \mathbf{x} \in \mathbb{X}_{k|k-1}^\ell \mid \mathbf{y}_k^\ell = \mathbf{g}_k^\ell(\mathbf{x}, \mathbf{v}), \mathbf{v} \in [\mathbf{v}] \right\}. \quad (13)$$

2. Between  $k$  and  $k + 1$ ,  
for  $r = 1$  to  $R_{\max}$  (number of round trips)

$$\mathbb{X}_{k|k}^{\ell,r} = \bigcap_{s \in \mathcal{C}(\{\ell\})} \mathbb{X}_{k|k}^{s,r-1} \quad (14)$$

3. Just before  $k + 1$

$$\mathbb{X}_{k|k}^\ell = \mathbb{X}_{k|k}^{\ell, R_{\max}}. \quad (15)$$

As for the centralized algorithm, the idealized distributed algorithm requires the evaluation of the direct and inverse images of a set by a function. Proposition 1 gives some conditions under which the distributed approach gives results similar to the centralized one.

**Proposition 1.** *Consider a WSN of  $L$  nodes represented by an entirely connected graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  of diameter  $d$ . Assume that at time  $k - 1$ ,  $\mathbb{X}_{k-1|k-1}^\ell = \mathbb{X}_{k-1|k-1}$  for all  $\ell \in \mathcal{V}$ . If the number of roundtrips  $R_{\max}$  satisfies  $R_{\max} \geq d$ , then one has at time  $k$*

$$\mathbb{X}_{k|k}^\ell = \mathbb{X}_{k|k} \quad (16)$$

for all  $\ell \in \mathcal{V}$ . ◇

*Proof.* Consider a vertex  $\ell \in \mathcal{V}$ . Since  $\mathbb{X}_{k-1|k-1}^\ell = \mathbb{X}_{k-1|k-1}$ , after the prediction step (12),  $\mathbb{X}_{k|k-1}^\ell = \mathbb{X}_{k|k-1}$ , where  $\mathbb{X}_{k|k-1}$  is provided by (7). The first correction step done at  $\ell$  involves only the measurement vector  $\mathbf{y}_k^\ell$  to get  $\mathbb{X}_{k|k}^{\ell,0}$ . After the first roundtrip, the estimate at  $\ell$  becomes

$$\begin{aligned} \mathbb{X}_{k|k}^{\ell,1} &= \bigcap_{s \in \mathcal{C}(\{\ell\})} \mathbb{X}_{k|k}^{s,0} \\ &= \bigcap_{s \in \mathcal{C}(\{\ell\})} \left\{ \mathbf{x} \in \mathbb{X}_{k|k-1}^s \mid \mathbf{y}_k^s = \mathbf{g}_k^s(\mathbf{x}, \mathbf{v}), \mathbf{v} \in [\mathbf{v}] \right\} \\ &= \bigcap_{s \in \mathcal{C}(\{\ell\})} \left\{ \mathbf{x} \in \mathbb{X}_{k|k-1} \mid \mathbf{y}_k^s = \mathbf{g}_k^s(\mathbf{x}, \mathbf{v}), \mathbf{v} \in [\mathbf{v}] \right\} \\ &= \left\{ \mathbf{x} \in \mathbb{X}_{k|k-1} \mid \mathbf{y}_k^{\mathcal{C}(\{\ell\})} = \mathbf{g}_k^{\mathcal{C}(\{\ell\})}(\mathbf{x}, \mathbf{v}), \mathbf{v} \in [\mathbf{v}] \right\}, \end{aligned}$$

where  $\mathbf{y}_k^{\mathcal{C}(\{\ell\})}$  and  $\mathbf{g}_k^{\mathcal{C}(\{\ell\})}(\mathbf{x}, \mathbf{v})$  are the vector and function consisting of the concatenation of all  $\mathbf{y}_k^s$  and  $\mathbf{g}_k^s(\mathbf{x}, \mathbf{v})$ , with  $s \in \mathcal{C}(\{\ell\})$ .

After a second roundtrip, the estimate at  $\ell$  becomes

$$\begin{aligned} \mathbb{X}_{k|k}^{\ell,2} &= \bigcap_{s \in \mathcal{C}(\{\ell\})} \mathbb{X}_{k|k}^{s,1} \\ &= \bigcap_{s \in \mathcal{C}(\{\ell\})} \left\{ \mathbf{x} \in \mathbb{X}_{k|k-1} \mid \mathbf{y}_k^{\mathcal{C}(\{s\})} = \mathbf{g}_k^{\mathcal{C}(\{s\})}(\mathbf{x}, \mathbf{v}), \mathbf{v} \in [\mathbf{v}] \right\} \\ &= \left\{ \mathbf{x} \in \mathbb{X}_{k|k-1} \mid \mathbf{y}_k^{\mathcal{C}(\{\ell\})} = \mathbf{g}_k^{\mathcal{C}(\{\ell\})}(\mathbf{x}, \mathbf{v}), \mathbf{v} \in [\mathbf{v}] \right\} \\ &= \left\{ \mathbf{x} \in \mathbb{X}_{k|k-1} \mid \mathbf{y}_k^{\mathcal{C}^2(\{\ell\})} = \mathbf{g}_k^{\mathcal{C}^2(\{\ell\})}(\mathbf{x}, \mathbf{v}), \mathbf{v} \in [\mathbf{v}] \right\}. \end{aligned}$$

Similarly, after  $R_{\max}$  roundtrips, one gets at  $\ell$

$$\mathbb{X}_{k|k}^{\ell, R_{\max}} = \left\{ \mathbf{x} \in \mathbb{X}_{k|k-1} \mid \mathbf{y}_k^{\mathcal{C}^{R_{\max}}(\{\ell\})} = \mathbf{g}_k^{\mathcal{C}^{R_{\max}}(\{\ell\})}(\mathbf{x}, \mathbf{v}), \mathbf{v} \in [\mathbf{v}] \right\}.$$

It is now enough to show that  $\mathcal{C}^{R_{\max}}(\{\ell\}) = \mathcal{V}$  in order to prove that  $\mathbb{X}_{k|k}^{\ell, R_{\max}} = \mathbb{X}_{k|k}$ . First, one has  $\mathcal{C}^{R_{\max}}(\{\ell\}) \subset \mathcal{V}$ . Assume now that there exists some  $k \in \mathcal{V}$  such that  $k \notin \mathcal{C}^{R_{\max}}(\{\ell\})$ . This means that  $k$  lies at a distance strictly larger than  $R_{\max}$  from  $\ell$ . Since the diameter  $d$  of  $\mathcal{G}$  is lower than  $R_{\max}$  the distance between two vertices is necessarily lower than  $d$ , which contradicts the initial assumption. Thus any  $k \in \mathcal{V}$  satisfies  $k \in \mathcal{C}^{R_{\max}}(\{\ell\})$  and  $\mathcal{C}^{R_{\max}}(\{\ell\}) = \mathcal{V}$ .

The result of Proposition 1 is not very surprising. It mainly states that when there are enough information exchanges between sensors, the distributed estimate converges at any sensor to the centralized estimate. What is more interesting is that the number of roundtrips needed for convergence depends only on the diameter of the graph associated with the WSN.

When  $R_{\max} < d$ , the situation is much more complex, since not all sensors will have access to all measurements (or to their contribution to the estimation of  $\mathbf{x}_k$ ). For the first roundtrips at time  $k+1$ , sensor  $\ell$  will have to broadcast information about  $\mathbb{X}_{k+1|k+1}^{\ell,r}$ , but also about  $\mathbb{X}_{k|k}^{\ell,r}$ , as long as  $\mathbb{X}_{k|k}^{\ell,r}$  has not converged to  $\mathbb{X}_{k|k}$ . Again, the diameter of the graph plays a crucial role. Further analysis is still required.

**Practical algorithm.** The implementation of the proposed idealized algorithm is done in a way similar to that of the centralized algorithm presented in [19]. In a most basic version of the algorithm, sets are represented by boxes, basic interval evaluations are performed for the prediction step and interval constraint propagation is done for the correction step. The advantage of this version is that it may readily be implemented on chips with reduced computational capabilities [27]. A more sophisticated version could involve description of sets using subpavings, a prediction step implemented using IMAGESP [19] and SIVIA [28] combined with interval constraint propagation for the correction step.



### 3 Applications

For the application part, a static localization problem for a single source is considered first. Then, the source will be moving, and the localization problem is cast into a problem of state estimation.

#### 3.1 Static source localization

The known location of the sensors is denoted by  $\mathbf{r}_\ell \in \mathbb{R}^2$ ,  $\ell = 1 \dots L$ . The unknown location of the source is  $\theta = (\theta_1, \theta_2)^T \in \mathbb{R}^2$ . The mean power  $\overline{P}_{\text{dB}}(d_\ell)$  (in dBm) received by the  $\ell$ -th sensor is described by Okumura-Hata model [29]

$$\overline{P}_{\text{dB}}(d_\ell) = P_0 - 10n_p \log \frac{d_\ell}{d_0}, \quad (17)$$

where  $n_p$  is the path-loss exponent (unknown, but constant),  $d_\ell = |\mathbf{r}_\ell - \theta|$ . The received power is assumed to lie within some bounds

$$P_{\text{dB}}(d) \in \left[ P_0 - 10n_p \log \frac{d}{d_0} - e, P_0 - 10n_p \log \frac{d}{d_0} + e \right], \quad (18)$$

where  $e$  is assumed known.

The RSS by sensor  $\ell = 1 \dots L$  may be rewritten as

$$y_\ell = h_\ell(\theta, A, n_p) v_\ell, \quad (19)$$

with

$$h_\ell(\theta, A, n_p) = \frac{A}{|\mathbf{r}_\ell - \theta|^{n_p}}, \quad A = 10^{P_0/10} d_0^{n_p}, \quad (20)$$

and  $v_\ell \in [v] = [10^{-e/10}, 10^{e/10}]$ . The noise is thus multiplicative in the normal domain. The parameter vector to be estimated is then  $\mathbf{x} = (A, n_p, \theta_1, \theta_2)^T$ .

**Distributed approach: interval constraint propagation.** At sensor  $\ell$ ,  $y_\ell \in [y_\ell]$  is measured. Some boxes  $[\theta]$ ,  $[A]$ , and  $[n_p]$  are assumed to be available, *a priori*, or as results transmitted by the other sensors to sensor  $\ell$ . The parameter vector has to satisfy the constraint provided by the RSS model

$$y_\ell - \frac{A}{|\mathbf{r}_\ell - \theta|^{n_p}} = 0. \quad (21)$$

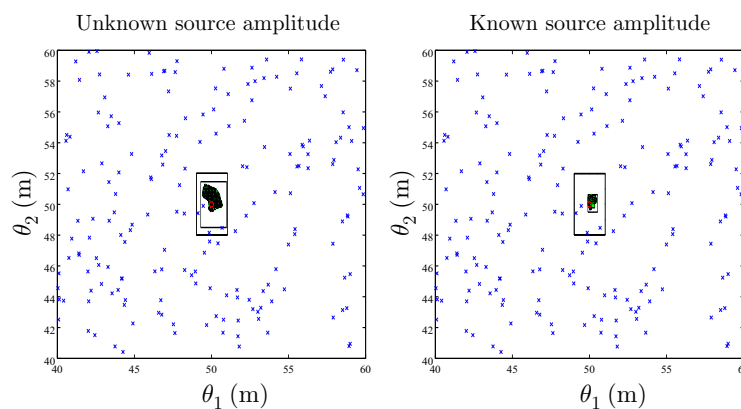
Using interval constraint propagation, it is possible to reduce the domains for the variables using (21). The contracted domains may be written as

$$\begin{cases} [y'_\ell] = [y_\ell] \cap \frac{[A]}{|\mathbf{r}_\ell - [\theta]|^{[n_p]}}, \\ [A'] = [A] \cap [y'_\ell] |\mathbf{r}_\ell - [\theta]|^{[n_p]}, \\ [n'_p] = [n_p] \cap (\log([A']) - \log([y'_\ell])) / \log(|\mathbf{r}_\ell - [\theta]|), \\ [\theta'_1] = [\theta_1] \cap \left( r_{\ell,1} \pm \sqrt{([A'] / [y'_\ell])^{2/[n'_p]} - (r_{\ell,2} - [\theta_2])^2} \right), \\ [\theta'_2] = [\theta_2] \cap \left( r_{\ell,2} \pm \sqrt{([A'] / [y'_\ell])^{2/[n'_p]} - (r_{\ell,1} - [\theta_1])^2} \right). \end{cases} \quad (22)$$

Sensor	68	741	954
Measurement	[9.303, 58.698]	[17.856, 112.664]	[18.644, 117.640]

**Table 1.** Example of measurements (static localization)

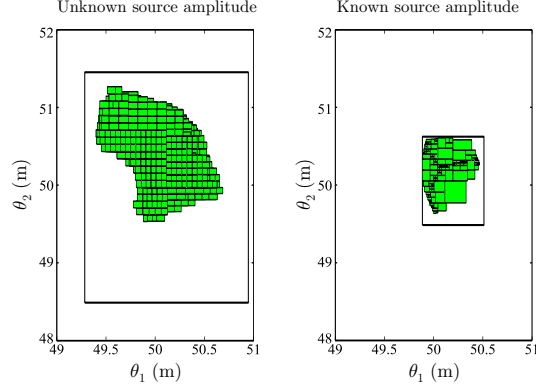
**Simulation results.** A network of  $L = 2000$  sensors randomly distributed over a field of  $100 \text{ m} \times 100 \text{ m}$  is considered. The source is placed at  $\theta^* = (50 \text{ m}, 50 \text{ m})$  and emits a wave with  $P_0 = 20 \text{ dBm}$ ,  $d_0 = 1 \text{ m}$ . The path-loss exponent  $n_p = 2$  is assumed to be constant over the field. The measurement noise is such that  $e = 4 \text{ dBm}$ . Table 1 provides some examples of the measurements which are available to the sensors.



**Fig. 4.** Projection of the solution on the  $(\theta_1, \theta_2)$ -plane

For 100 realizations of the sensor field, data have been simulated with (18). To limit computational load, only sensors such that  $y_\ell > 10$  participate to localization. The initial search box for  $\mathbf{p}$  is taken as  $[0, 100] \times [0, 100] \times [50, 200] \times [2, 4]$  in a first scenario, where  $A$  (or  $P_0$ ) is assumed unknown. In a second scenario,  $A$  is assumed perfectly known. For the distributed approach, five cycles in the sensor network are performed.

The two proposed techniques are compared to localization by a closest point approach (CPA), which searches for the index of the sensor with the largest RSS  $\ell_{\text{CPA}} = \arg \max_\ell y_\ell$  and uses the location of this sensor  $\hat{\theta}_{\text{CPA}} = \mathbf{r}_{\ell_{\text{CPA}}}$  as an estimate for  $\theta^*$ . This technique, albeit it is not the most efficient [5], performs well for dense sensor networks, as here. Point estimates for  $\theta^*$  are evaluated as  $\hat{\theta}_{\text{C}} = \text{mid}([\text{proj}_\theta \bar{\mathbb{P}}])$ , the midpoint of the smallest box containing the projection of  $\bar{\mathbb{P}}$  onto the  $\theta$ -plane in the centralized approach and as the center of the projection onto the  $\theta$ -plane of the solution box  $[\mathbf{p}]$ ,  $\hat{\theta}_{\text{D}} = \text{mid}(\text{proj}_\theta [\mathbf{p}])$ , in the distributed approach.



**Fig. 5.** Zoom of the projection of the solution on the  $(\theta_1, \theta_2)$ -plane

Figures 4 and 5 provide typical solutions obtained using a centralized and distributed localization algorithm. The centralized algorithm involves set description using subpavings, whereas the distributed one only uses boxes, to limit the amount of information exchanged between sensors.

Figure 6 presents the histogram of the  $L_2$  norm of the difference between  $\theta^*$  and its estimates ( $\hat{\theta}_{\text{CPA}}$ ,  $\hat{\theta}_{\text{D}}$ , and  $\hat{\theta}_{\text{C}}$ ) provided by the three techniques previously described. The centralized approach performs better than the distributed one, but the distributed approach provides a reasonable estimate at a much lower computation and transmission cost. Both techniques outperform CPA, the performances of which do not depend on whether  $A$  is known.

### 3.2 Source tracking

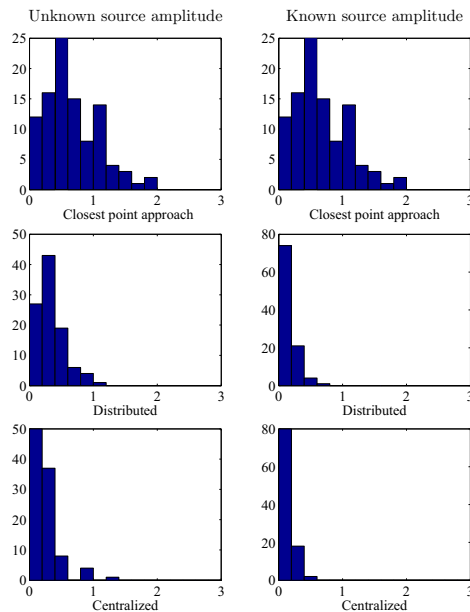
In this part, the source is assumed to be moving.  $A$  and  $n_p$  are now known. The state vector is taken as

$$\mathbf{x}_k = (\theta_{1,k}, \theta_{2,k}, \phi_{1,k}, \phi_{2,k}, \theta_{1,k-1}, \theta_{2,k-1}, \phi_{1,k-1}, \phi_{2,k-1})^T \quad (23)$$

where  $(\phi_1, \phi_2)$  represents the speed of the source. This extended state vector is considered, as it allows to estimate  $(\phi_{1,k}, \phi_{2,k})$ .

**Model.** The following uncertain linear state equation is considered to determine the evolution with time of  $\mathbf{x}_k$

$$\begin{pmatrix} \theta_{1,k} \\ \theta_{2,k} \\ \phi_{1,k} \\ \phi_{2,k} \\ \theta_{1,k-1} \\ \theta_{2,k-1} \\ \phi_{1,k-1} \\ \phi_{2,k-1} \end{pmatrix} = \begin{pmatrix} \mathbf{I}_4 & \mathbf{0}_4 \\ \mathbf{I}_4 & \mathbf{0}_4 \end{pmatrix} \begin{pmatrix} \theta_{1,k-1} \\ \theta_{2,k-1} \\ \phi_{1,k-1} \\ \phi_{2,k-1} \\ \theta_{1,k-2} \\ \theta_{2,k-2} \\ \phi_{1,k-2} \\ \phi_{2,k-2} \end{pmatrix} + T \cdot \begin{pmatrix} \phi_{1,k-1} \\ \phi_{2,k-1} \\ w_1 \\ w_2 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad (24)$$



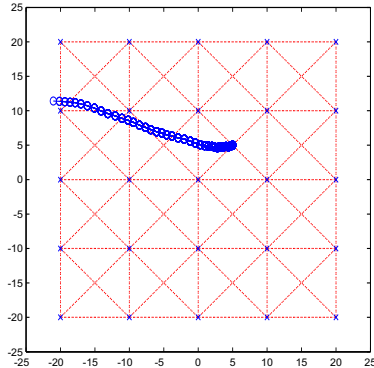
**Fig. 6.** Histograms of estimation error (in meters) for  $\theta$  (100 realizations of the sensor field)

Since the inputs are unknown, they are considered as bounded state perturbations. Thus,  $w_1 \in [w]$  and  $w_2 \in [w]$ .

**Interval constraint propagation.** Interval constraint propagation is used at the correction step. From (21) and (24), one gets the contracted domains at node  $\ell$

$$\left\{ \begin{array}{l} [y'_{\ell,k}] = [y_{\ell,k}] \cap \frac{A}{[\mathbf{r}_{\ell} - [\theta_k]]^{[n_p]}}, \\ [\theta'_{1,k}] = [\theta_{1,k}] \cap \left( r_{\ell,1} \pm \sqrt{\left( A / [y'_{\ell,k}] \right)^{2/n_p} - (r_{\ell,2} - [\theta_{2,k}])^2} \right), \\ [\theta'_{2,k}] = [\theta_{2,k}] \cap \left( r_{\ell,2} \pm \sqrt{\left( A / [y'_{\ell,k}] \right)^{2/n_p} - (r_{\ell,1} - [\theta_{1,k}])^2} \right). \\ \\ [\phi'_{1,k}] = [\phi_{1,k}] \cap \left( \frac{[\theta'_{1,k}] - [\theta_{1,k}]}{T} + T[w] \right), \\ [\phi'_{2,k}] = [\phi_{2,k}] \cap \left( \frac{[\theta'_{2,k}] - [\theta_{2,k}]}{T} + T[w] \right). \end{array} \right. \quad (25)$$

Each sensor will perform this constraint propagation before transmitting its updated estimate to its neighbours.



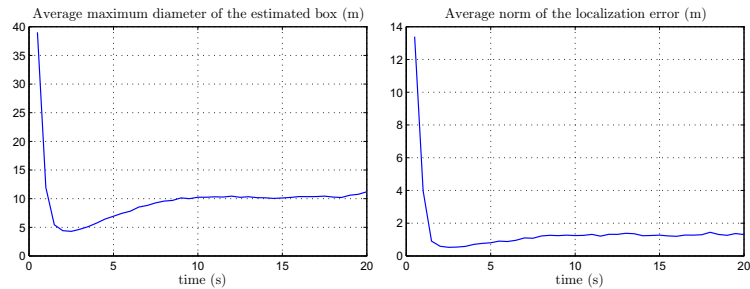
**Fig. 7.** Trajectory of the source (o); each sensor is represented by a cross (x), distances are in meters

**Results.** Now, a field of  $50\text{ m} \times 50\text{ m}$  is considered, with its origin at the center. A WSN of  $L = 25$  sensors with communication range of  $15\text{ m}$  is spread over this field. The source is placed at  $\theta^* = (5\text{ m}, 5\text{ m})$ , with characteristics  $\bar{P}_0 = 20\text{ dBm}$ ,  $d_0 = 1\text{ m}$ . The measurement noise is such that  $e = 4\text{ dBm}$ . The path-loss exponent is  $n_p = 2$ , assumed constant over the field. The sampling time is  $T = 0.5\text{ s}$  and  $[\mathbf{w}] = [-0.5, 0.5]^2\text{ m}\cdot\text{s}^{-2}$ . Figure 7 illustrates the connectivity of the considered regular WSN and a typical trajectory followed by the source.

The simplest algorithm implementation presented in Section 2.2 has been considered: sets are represented by boxes, simple image evaluations using inclusion functions are performed and correction is done by interval constraint propagation. This limits the amount of information to be exchanged between sensors and the computational effort. The localization performance using this algorithm is depicted in Figure 8 for 100 realizations of the source trajectory. The average width of the solution box (left part of Figure 8) provided at each time instant decreases very quickly before reaching a floor slightly higher than the minimum width and increases again after about  $18\text{ s}$ . At the beginning, the source is close to the middle of the field and many sensors participate to the localization. When the sensor moves near the limits of the field, the number of involved sensors decreases and as a result the localization accuracy worsens. This effect is even more important when the source moves outside the field. A similar behavior is seen for the average norm of the localization error taking the center of the solution boxes at each time instant as estimate. The convergence is quite fast and the number of round trips has only a very limited impact on the convergence of the algorithm.

## 4 Conclusions

In this paper, we have considered distributed bounded-error state estimation applied to the problem of source tracking with a network of wireless sensors.



**Fig. 8.** Width of the box  $[\theta_{1,k}] \times [\theta_{2,k}]$ , and norm of the localization error when the estimate is taken as the center of the solution box (average over 100 random paths followed by the source)

Estimation is performed in a distributed context, *i.e.*, each sensor has only a limited amount of measurements available. A guaranteed set estimator is put at work.

There is still large space for improvements in the considered problem. First, convergence properties have to be more carefully studied. In particular, more general conditions under which the distributed solution coincides with the centralized one have to be determined. This type of problem is partly addressed in [30, 31]. Robustness to outliers and network optimization for optimal estimation have also to be considered. Another challenging application would be the distributed estimation, *e.g.*, in a team of robots.

## References

1. R. Kay and F. Mattern. The design space of wireless sensor networks. *IEEE Wireless Communications*, 11(6):54–61, 2004.
2. T. Haenselmann. *Sensornetworks. GFDL Wireless Sensor Network textbook*. 2006. [http://www.informatik.uni-mannheim.de/haensel/sn\\_book](http://www.informatik.uni-mannheim.de/haensel/sn_book).
3. N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal. Locating the nodes. *IEEE Signal Processing Magazine*, 22(4):54–69, 2005.
4. A. H. Sayed, A. Tarighat, and N. Khajehnouri. Network-based wireless location. *IEEE Signal Processing Magazine*, 22(4):24–40, 2005.
5. X. Sheng and Y. H. Hu. Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. *IEEE Transactions on Signal Processing*, 53(1):44 – 53, 2005.
6. P. Kontkanen, P. Myllymäki, T. Roos, H. Tirri, K. Valtonen, and H. Wettig. Probabilistic methods for location estimation in wireless networks. In R. Ganesh, S. Kota, K. Pahlavan, and R. Agustí, editors, *Emerging Location Aware Broadband Wireless Adhoc Networks*. Kluwer Academic Publishers, 2004.
7. F. Gustafsson and F. Gunnarsson. Mobile positioning using wireless networks. *IEEE Signal Processing Magazine*, 22(4):41–53, 2005.
8. S. Gezici, Z. Tian, G. B. Giannakis, H. Kobayashi, A. F. Molish, H. V. Poor, and Z. Sahinoglu. Localization via ultra-wideband radios. *IEEE Signal Processing Magazine*, 22(4):70–84, 2005.

9. M. G. Rabbat and R. D. Nowak. Decentralized source localization and tracking. In *Proc. ICASSP*, 2004.
10. A. O. Hero III and D. Blatt. Sensor network source localization via projection onto convex sets (POCS). In *Proceedings of ICASSP*, 2005.
11. R. E. Moore. *Interval Analysis*. Prentice-Hall, Englewood Cliffs, NJ, 1966.
12. L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. *Applied Interval Analysis*. Springer-Verlag, London, 2001.
13. R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the AMSE, Part D, Journal of Basic Engineering*, 82:35–45, 1960.
14. A. Gelb. *Applied Optimal Estimation*. MIT Press, Cambridge, MA, 1974.
15. P. Terwiesch and M. Agarwal. A discretized non-linear state estimator for batch processes. *Computers and Chemical Engineering*, 19:155–169, 1995.
16. M. Pitt and N. Shephard. Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, 94(446):590–599, 1999.
17. F. C. Schweppe. Recursive state estimation: unknown but bounded errors and system inputs. 13(1):22–28, 1968.
18. D. Maksarov and J. P. Norton. State bounding with ellipsoidal set description of the uncertainty. 65(5):847–866, 1996.
19. M. Kieffer, L. Jaulin, and E. Walter. Guaranteed recursive nonlinear state bounding using interval analysis. *International Journal of Adaptive Control and Signal Processing*, 6(3):193–218, 2002.
20. H. Kwakernaak and R. Sivan. *Linear Optimal Control System*. John Wiley–Interscience, 1974.
21. R. Hermann and A. J. Krener. Nonlinear controllability and observability. *IEEE trans. Automatic Control*, 22(5):728–740, 1977.
22. J. Speyer. Computation and transmission requirements for a decentralized linear-quadratic-gaussian control problem. *IEEE Trans. Automatic Control*, 24(2):266–269, April 1979.
23. S.-S. Lin and Huay Chang. An efficient algorithm for solving distributed state estimator and laboratory implementation. In *ICPADS '05: Proceedings of the 11th International Conference on Parallel and Distributed Systems (ICPADS'05)*, pages 689–694, Washington, DC, USA, 2005. IEEE Computer Society.
24. A. Ribeiro, G. B. Giannakis, and S. I. Roumeliotis. SOI-KF: Distributed Kalman filtering with low-cost communications using the sign of innovations. *IEEE Trans. Signal Processing*, 54(12):4782–4795, December 2006.
25. F. Harary. *Graph Theory*. Addison-Wesley, Reading, MA, 1994.
26. B. Bollobás. *Modern Graph Theory*. Springer-Verlag, New-York, 1998.
27. S. Piskorski, L. Lacassagne, M. Kieffer, and D. Etiemble. Efficient 16-bit floating-point interval processor for embedded systems and applications. In *Proc. 12th GAMM - IMACS Int. Symp. on Scientific Computing, Computer Arithmetic and Validated Numerics (SCAN 2006)*, pages 23–23, 26–29 Sept. 2006.
28. L. Jaulin and E. Walter. Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica*, 29(4):1053–1064, 1993.
29. Y. Okumura, E. Ohmori, T. Kawano, and K. Fukuda. Field strength and its variability in VHF and UHF land-mobile radio service. *Rev. Elec. Commun. Lab.*, 16:9–10, 1968.
30. M. Yokoo. *Distributed Constraint Satisfaction: Foundations of Cooperation in Multi-Agent Systems*. Springer-Verlag, Berlin, 2001.
31. R. Bejar, C. Fernandez, M. Valls, C. Domshlak, C. Gomes, B. Selman, and B. Krishnamachari. Sensor networks and distributed CSP: Communication, computation and complexity. *Artificial Intelligence Journal*, 161(1-2):117–148, 2005.